CFD in Support of Wind Tunnel Testing for Aircraft/Weapons Integration

William L. Sickles
Aerospace Testing Alliance, Arnold Engineering Development Center (AEDC), Arnold AFB, TN
bill.sickles@arnold.af.mil

Abstract

Integrating data and computations using AEDC semiempirical and CFD techniques has provided vital support for integrating weapons with the Joint Strike Fighter (JSF) and the Joint Unmanned Combat Air System (J-UCAS). The techniques and the processes used are described. Computational results were obtained using HPC assets, and when possible, the results were validated against wind tunnel data. The success of these validations has led to high confidence in the computational results and also to the use of computational results to provide data outside the tunnel operating envelope. This has led in turn to the elimination of tests and saved the two programs millions of dollars. In summary, the subject computations: 1) provided information for pretest planning; 2) corrected errors in wind tunnel data reduction and results; and 3) obtained results for configurations that could not be obtained in the wind tunnel because of physical limitations of the support systems as well as limitations of the tunnel operating envelope. Computations and analysis performed for the two programs have demonstrated that the Integrated Test and Evaluation process does reduce cost and risk.

1. Introduction

Computational fluid dynamics (CFD) applications to support wind tunnel tests at the Arnold Engineering Development Center (AEDC) have increased each year for over fifteen years. A myriad of aircraft programs have been supported including the F-15E, F-16, F-18E/F, B-1B, B-2, F-22, JSF, and UCAV, with associated weapon systems (JDAM, JSOW, WCMD, JASSM, SDB, and PLOCAAS). Over the past two years, AEDC has computed thousands of viscous, steady-state solutions to augment weapons integration wind tunnel testing. The CFD solutions have involved solutions for aircraft with a variety of loadouts, weapons (or stores) alone, as well as fully viscous, time-accurate weapon trajectory simulations. With increased emphasis on reducing cost, schedule, and risk, flight system programs are relying on CFD to supplement and complement wind tunnel data. CFD is no longer just a “nice thing to have”; it is an integral part of the development process, and therefore computational results must adhere to the development schedule. Therefore, the results must be provided in a timely manner. This time constraint places an increased demand on AEDC to improve process and flow-solver efficiencies and to utilize modern computational sources to yield the turnaround necessary. The level of computational support to weapons integration has pushed the limits of the AEDC High-Performance Computing (HPC) Distributed Center (DC) computer resources to provide these time-critical results. The need for additional capacity was the original basis for AEDC’s successful proposal for an HPC Challenge Project.

Programs currently using the Challenge Project and the AEDC DC resources are the Joint Strike Fighter (JSF) and the Joint Unmanned Combat Air System (J-UCAS), formerly UCAV. Both programs are using CFD to supplement weapons integration testing. Hundreds of viscous flow-field solutions have been computed on both vehicles in the last nine months, helping to define the required wind tunnel tests and scope, while providing aircraft and weapons flow-field information for analysis of results. It should be emphasized that HPC resources support not just these programs, but a variety of flight systems and facility technology efforts. These other programs are primarily taking advantage of local DC and remote Major Shared Resource Center (MSRC) resources outside the allotment of Challenge Project.

For the JSF program, time-accurate trajectory simulations of separating stores were computed for analysis of both internal and external carriage releases. In addition, computations have played a vital role in another aspect of weapons integration, i.e., providing store carriage loads data for airframe structural analysis. Store carriage loads were computed for a variety of loadout configurations at flow conditions inside and outside the tunnel envelope.

0-7695-2259-9/04 $20.00 © 2004 IEEE

For the J-UCAS program, flow-field computations were performed prior to wind tunnel testing to 1) determine wind tunnel wall interference effects on the store separation data, 2) analyze weapons bay unsteadiness, 3) analyze weapons bay spoiler candidates and their effects on weapons bay dynamics, and 4) analyze quasi-steady store separation with engineering methods.

Because the J-UCAS program is in the competitive phase, it is not possible to discuss any of the computational or experimental results. Although the JSF program is past its competitive phase, all the results that have been computed are Lockheed proprietary or critical technology that cannot be placed in the open literature. The sensitivity of the geometry, weapons loadouts, flow conditions, and any computational results severely limits what can be discussed in this forum. Therefore, this paper is limited in results and primarily emphasizes weapon integration procedures and processes at AEDC and how CFD is utilized in that process. Also discussed is the impact that the computational effort, which is supported by HPC assets, is having on these two development programs.

2. Methods for Separation Analysis

Over the past ten years, algorithm improvements to the AEDC XAIR (now NXAIR) code have significantly reduced the computational time required to perform purely computational time-accurate moving-body simulations. The computational time for time-accurate store separation analysis has decreased by an order of magnitude[1]. The code has also been modified to take advantage of the parallel architecture of modern multiprocessor machines such as those machines provided by many HPC sites. Coupling these improvements with improvements in interactive grid generation has significantly improved the productivity of the overall computational simulation process. Codes that can efficiently produce time-accurate moving-body simulations are becoming important practical tools for predicting store separations, thereby complementing the wind tunnel data and engineering methods. These improvements represent a vital step towards addressing the productivity and cost issues associated with performing this type of analysis. The time-accurate simulations can now be routinely applied to problems that were considered too computationally intensive ten years ago; such simulations are now a practical alternative and complementary tool to the workhorse quasi-steady engineering approaches.

The time-accurate approach is universally considered to be the most accurate method for simulating the store separation event. It captures the temporal variation of the flow field caused by the mutual interaction between the moving store(s) and the parent aircraft. Capturing these temporal variations is particularly important, and even necessary, for simulating more complex problems such as the simultaneous and staged releases of multiple stores and the release of stores from weapons bays at certain flight conditions.

However, the time-accurate simulation of the store separation event is still computationally intensive and does not offer the opportunity to quickly perform parametric studies such as variations in the ejector model and the store inertial properties. These parametric studies are easily done with the quasi-steady engineering methods. Previous investigations have shown that quasi-steady engineering methods give accurate results for a wide class of separation problems and particularly for release of a single, external store from a pylon[2,3]. Because of their unique advantages, both the quasi-steady engineering methods and the time-accurate simulations will be used for the near future and should be used to complement each other.

Quasi-steady engineering methods have been widely used to perform online and offline store separation analysis from wind tunnel data[4,5]. They have played a vital role in reducing wind tunnel cost by reducing the volume of data required to accurately predict the store separation event. These engineering methods are not limited to the utilization of wind tunnel data. CFD has been used to supply some or all of the information required by the engineering approach by simulating the wind tunnel data[2,6–8]. This information can be the freestream flow field beneath the parent aircraft, the store freestream loads, or the store carriage (and near carriage) loads. The use of both CFD and wind tunnel data yields a hybrid approach where the existing wind tunnel data, e.g., freestream store loads, can be utilized to reduce the number of CFD solutions required and, conversely, the CFD can be utilized to reduce the wind tunnel data requirements.

AEDC has extensive experience in the application of both time-accurate moving-body computations[3] and engineering quasi-steady methods in simulating the store separation event[5-8]. It should be noted that the use of both approaches for a given program reduces the risk of errors and improves the quality of the end product. Results from both purely computational time-accurate and hybrid quasi-steady approaches have been successfully validated against wind tunnel data and high-quality flight data[2,3].

It is assumed that a store separates from the parent aircraft in rigid-body motion. Therefore, its trajectory is determined by a well-known system of ordinary differential equations that model such motions. In these equations of motion, temporal variations of state variables (position, orientation, velocity, and rotation rates) are related to integrated forces and moments acting on the body. Specifying these loads constitutes the greatest challenge for advancing the trajectory in time.

Consider the difficulty in meeting this challenge with a truly time-accurate approach. In a computational setting, such an approach requires the computation of the time-varying flow coordinated with each movement of the separating body. With changes in the flow there would be time-varying aerodynamic forces and moments distributed over the body surface. At a given time, these would have to be integrated and combined with nonaerodynamic forces and moments (e.g., from ejectors) to obtain resultant loads. These loads would be used to drive the motion, and thus the cycle would continue. Formerly, such a computational simulation required large amounts of computer resources. Significant progress has been made in recent years at AEDC in reducing the amount of computational resources required to perform time-accurate moving-body simulations. Although order-of-magnitude reductions have been made, the time-accurate CFD approach does not allow for quick recalculation of the trajectory to perform parametric studies such as ejector and mass property variation. In an experimental setting, the time-accurate approach can be simulated with free-drop testing. However, free-drop testing is costly in models and test time and is rarely used as a test technique. Because of the shortcomings of the experimental and computational time-accurate approaches, engineering methods were developed and continue to be used to specify trajectory loads[4].

3. Engineering Methods

Engineering methods all rely on a type of quasi-steady approximation. For example, a store can be positioned in the aircraft flow field and flow allowed to develop. Then aerodynamic loads are specified by modifying measured loads with a first-order correction to account for the fact that flow is allowed to develop over a finite time interval. In the wind tunnel, this process is called the dual-sting support or captive-trajectory support technique.

Yet another level of approximation has been successfully introduced in a class of methods in which the store is superimposed on a flow field and the loads are extracted from a database. One option for this approach involves generating a map of store loads for a range of orientations and positions in a region beneath the parent aircraft. Then loads for a store in a given orientation and position are interpolated from the load map. This method is called the store-loads survey approach. Another option is to generate only a map of the flow field over a region beneath the aircraft. Then, with the store in a given orientation and location, loads along the store are specified from a correlation between quasi-steady loads and flow conditions in the map. This method is called the flow-field survey approach.

Calculation of store loads using the flow-field survey approach does not account for mutual interference between the store and the aircraft. Mutual interference effects decay rapidly with distance from carriage and are significant only within the immediate vicinity of the parent aircraft[5]. As discussed later, carriage loads can be obtained and used to correct loads that do not include mutual interference effects at and near carriage.

It is important to note that the steady flow fields required in the engineering methods can be obtained experimentally or computationally. AEDC maintains an extensive database of freestream experimental store loads that is readily available. Indeed, an important aspect of this engineering methods approach is that this database can be utilized and CFD used to generate all other required data except freestream store loads. This marriage of CFD and engineering methods has previously been employed at the AEDC[2,5-8] to successfully predict store separation from fighter aircraft at transonic speeds.

3.1. Trajectory Generation Program.

Separation trajectories to be obtained using engineering methods are predicted using the Flow-field Loads Influence Prediction Trajectory Generation Program (FLIP TGP). The FLIP TGP is a follow-on version of the Flow-angle Trajectory Generation Program (FLOW TGP)[5]. Separation trajectories are computed in a marching process starting at carriage. At a given time, the aerodynamic and nonaerodynamic loads acting on a store are determined. The loads are used in the rigid-body equations of motion to determine the time derivatives of the store state variables (position, orientation, velocity, and rotation rates). The known derivatives permit the state variables to be extrapolated over a small time step to predict the change in store position and orientation. The loads acting on the store at the new position and orientation are then determined. The process is repeated until the store is moved for a specified time or a collision is detected.

3.2. Load Determination Procedure.

For the present engineering methods application, store aerodynamic loads were calculated according to the flow-field survey approach using the Missile Distributed Airloads (MDA) Code. In the flow-field survey approach, loads are determined by generating a flow-field map over a grid situated beneath the parent aircraft. Mapping of the flow field involves determination of local flow variables (Mach number, dynamic pressure, and velocity components) on the specified grid without the store present. Experimentally, this is accomplished by moving a probe from one grid location to another and making pressure measurements at each station. Computationally, this involves interpolating a CFD flow-field solution onto the specified grid. As explained earlier, these flow conditions are correlated to quasi-steady store loads.

The MDA Code, which is an upgraded version of the Interference Distributed Loads (IDL) Code[9], is a semiempirical program capable of predicting aerodynamic loads for a wide range of missile and bomb configurations at subsonic, transonic, and supersonic Mach numbers. The MDA Code may be operated either as a stand-alone program or as part of the FLIP TGP.

The MDA Code uses a component buildup approach to calculate the loads on a complete configuration. In this approach the store can be superimposed on either a uniform (freestream) or a specified nonuniform flow field. In both cases, force coefficients and centers of pressure are predicted for each of a number of body segments and for each fin. The component loads are corrected for interaction between the components, including vortex effects. Total loads are determined by summation of the component loads. For nonuniform-flow-field applications, local flow conditions are used to determine the loads on each component.

The MDA Code can be used for FLIP TGP applications in one of two ways. First, the MDA Code can be used to directly calculate total aerodynamic loads acting on the store in the aircraft flow field. Alternatively, the accuracy of these predictions can be improved if freestream aerodynamic store data are available. These data are used to generate a "uniform-flow component" of the total store loads as follows: The local flow conditions are averaged along the store. Then the freestream data are interpolated for the averaged conditions to produce the uniform-flow component of the total store loads. Next, the MDA Code is used to generate a "nonuniformity correction" for the total store loads. Such a correction is given by the difference between the direct calculation of loads with the MDA Code and the uniform-flow component of store loads.

If store freestream aerodynamic data are available, it is possible to improve store load predictions further by calibrating the MDA Code. The calibration process consists of analyzing freestream prediction errors to identify specific areas of discrepancy and correcting the appropriate MDA Code methods accordingly. The result is a modified version of the MDA Code, which should more accurately model the aerodynamics of the store in the aircraft flow field. In the present work, experimental freestream data from previous tests on similarly scaled stores were used as a basis for calibration of the MDA Code. With CFD solutions, the distribution of loads over the entire store can be evaluated and can be used to further calibrate the MDA Code.

Mutual interference corrections are determined on the basis of comparisons between MDA Code predictions and CFD or experimentally generated store loads at and near the carriage position. Typically, the near carriage loads are determined at two positions below carriage displaced a distance of one and two store diameters. The required MDA Code predictions are generated using the MDA Code with parent aircraft flow-field information. Differences between the MDA predictions and store loads predicted at the same locations on the basis of CFD solutions are assumed to be attributable to mutual interference between the moving store and the parent vehicle.

4. Time-Accurate CFD Methodology

The moving-body capability at AEDC relies on the chimera overset grid approach to compute the time-accurate flow fields and the unsteady aerodynamic forces and moments. The basic chimera overset-mesh procedure[10] allows the modeling of a complex body using relatively simple overlapping meshes, where each mesh represents a component of the body. The solution of the composite mesh is made continuous by the intercommunication among the individual meshes. Another advantage of the chimera approach is that it is easily adapted to parallel processing, where the computation of the flow field for each mesh can be spawned to different processors.

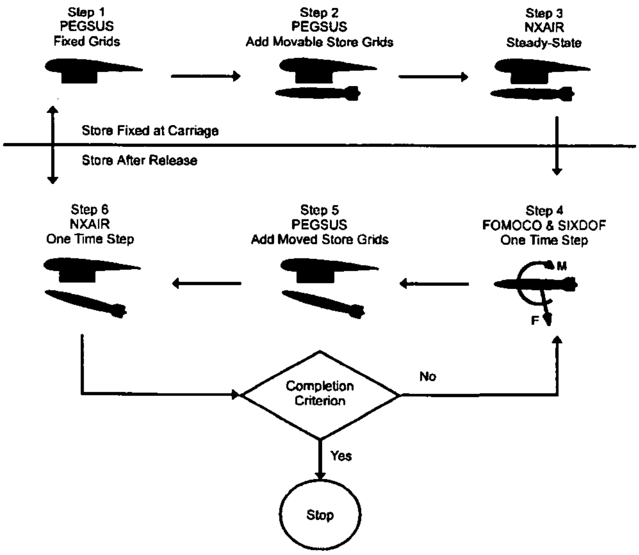

Figure 1. Time-Accurate Moving-Body (Store) Trajectory Prediction Process

The overall procedure for predicting body motion with time-accurate computations is given in Figure 1. Details of the process and the codes used are given in Reference 11. For completeness, a brief description of the codes and process is given here. The process is script driven and is composed of a series of codes that perform unique functions. Four codes comprise the heart of the solution process. NXAIR solves the fluid dynamic equations, PEGSUS defines the intergrid communications, FOMOCO computes the store loads, and SIXDOF solves the rigid-body equations of motion. The initial conditions for the trajectory are generated for the static carriage configuration, also using the NXAIR flow solver. It should be noted that NXAIR is also used to compute the static flow-field grid beneath the aircraft without the active store present, and the additional near carriage loads required for the trajectory simulations are computed using the quasi-steady engineering method.

4.1. NXAIR.

The code used in the present methodology for solving the fluid dynamic partial differential equations is the NXAIR code (formerly called XAIR). NXAIR is a node-centered code based upon the chimera grid structure for solving the full nonlinear Navier-Stokes equations, or the Euler subset. Numerous enhancements have made NXAIR an ideal code for solution of moving-body or unsteady flow problems. Recent algorithm improvements allow choosing time steps commensurate with that required to capture the physics, rather than that required to satisfy a numerical stability limit, thus greatly reducing the computer resource requirements for solving moving-body problems.

NXAIR employs a Harten/Lax/van Leer/Einfeldt (HLLE) upwind differencing scheme[1,12,13] for the right-hand side of the equations and achieves third-order spatial accuracy through a monotone upstream scalar conservation law (MUSCL) scheme[14]. First- or second-order time-accurate integration is achieved through a global Newton iteration algorithm. Jacobi or Gauss-Seidel subiterations are used at each time step to reduce the factorization error. This has the effect of increasing the allowable time step size by an order of magnitude or more, over the original alternating-direction-implicit (ADI) scheme. An added benefit of the increased time step size is that fewer calculations of interpolation stencils and holes for the overset grids are required. Those calculations usually constitute a significant portion of the overall solution time; therefore, any improvement that reduces the number of time steps is doubly significant.

Inviscid flux Jacobians are calculated using the Steger-Warming flux vector splitting scheme. Viscous fluxes (computed using spatial central differencing) in each of the three coordinate directions can be turned on or off independently, allowing for either full-, or thin-layer viscous calculations. A variety of turbulence transport models are available including the K-ε[15], Spalart-Allmaras[16] (S-A), and Menter's shear stress transport (SST)[17]. The Baldwin-Lomax algebraic model and the Wilcox K-ω model are also available. The K-ε, S-A, and SST transport models are solved separately from the flow equations, but are coupled within the Newton algorithm for efficiency. Wall functions, detached eddy simulation (DES), and multi-scale hybrid modifications are available for several of the aforementioned turbulence models. The same differencing schemes are used for the transport algorithms and the basic flow equations. Most boundary conditions are applied quasi-implicitly within the Newton subiteration proecess.

In addition to the aforementioned algorithm changes, the code has been "parallelized" using message passing interface (MPI) to take advantage of the architecture of new multiprocessor computers. A coarse-grain parallelization is used in which the code spawns the computations of individual meshes to as many processors as are allocated by the user. Improved scalability is achieved by attempting to uniformly distribute grid points amongst the meshes. This method of load balancing is done when generating the grid system and is preferred, but not totally achievable. Additional processor load balancing is gained in the code by grouping smaller meshes. Recent versions of the code also incorporate a multigrid strategy.

4.2. PEGSUS.

The PEGSUS code is an integral part of both the static and moving-body portions of the process as shown by the flow chart in Figure 1. The code positions the moving grids in the proper location with respect to the nonmoving, or fixed, grids, performs the hole-cutting function, and constructs the interpolation stencils among the interacting grids. Version 4.0 of PEGSUS[18] includes enhanced capabilities, namely the ADD and restart features (see below), which are useful for application to moving-body problems. These features are used in the following manner: First, a separate PEGSUS run is made for the baseline grid configuration that does not contain the moving grids. From PEGSUS, restart files are generated that contain the interactions among all grids except the moving-body grids. This eliminates the necessity and corresponding time required to regenerate these links if they are not altered by the later addition of the moving grids. Next, the ADD feature is used to generate the new grid system with the moving grids from baseline restart files. Initially, the ADD feature is used to rotate the moving meshes to the orientation of the static geometry (usually the carriage orientation). During the next and subsequent time steps, the ADD input is modified to account for the translation and rotation of the moving grids, and another grid system is regenerated from the baseline restart files. Although the ADD and restart features are time savers, the PEGSUS stage is still a significant part of the process.

The ADD feature is useful, but it is only applicable for an aircraft with a fixed reference system. For nonmanuevering aircraft, the aircraft remains fixed relative to its initial reference system, and the store's movement is tracked relative to the fixed reference system. For manuevering aircraft, this is no longer the case. When the aircraft goes into a nose-over or pull-up manuever, the aircraft is moving with respect to its initial reference system. The ADD feature is no longer an option because the movement of both the aircraft and the store must be adjusted, and a full PEGSUS run needs to be computed.

Previous versions of the flow solver required such small time steps that updating of the hole cutting and the interpolation information was not necessary at each time step. With the larger time step possible with NXAIR, updates are required after each time step. The efficiency of this step has been improved by parallelization of PEGSUS through compiler directives.

4.3. FOMOCO.

When using chimera overset grid systems, forces and moments are required for a collection of overlapping solid walls. Simple integration of every wall surface results in multiple bookkeeping of forces when summing all components for a complete configuration. The FOMOCO[19] code computes force and moment coefficients on a collection of overset surfaces by accounting for the overlapped zones. The grids which overlap are prioritized according to grid density (or user specification), and holes are cut via "i-blanking" in grids of lower priority, i.e., certain points are flagged with zero factors to eliminate them from the current computation. These newly i-blanked grid surfaces are reconnected with unstructured grid strips, called zippers. The code has been modified at AEDC to allow direct reading of the pressure and shear stress components, thus producing the ability to determine load coefficients when using wall functions.

4.4. SIXDOF.

Moving-body problems can be classified as prescribed, constrained, or free motion. All involve unsteady flow, and best results require time-accurate computations. In the case of prescribed motion, the influence of the flow field on the body motion is ignored, and the body moves during a given time interval along a predetermined path. In constrained motion, the aerodynamic forces actuate the body, but one or more of the degrees of freedom are neglected (e.g., the translations are prescribed, but the rotations are free to vary according to the aerodynamic forces). For free motion, however, the influence of the flow field on the body, in terms of aerodynamic forces and moments and gravity, totally determines the translational and rotational movement of the body. Thus, at each discrete time within the flow solution process, the current forces and moments must be determined; these forces and moments, in turn, are input to the rigid-body equations of motion. At AEDC the six degree-of-freedom (SIXDOF) code solves these equations and provides body and grid movements and velocities back to the flow solver for computation at additional time steps. A different SIXDOF code is used for the quasi-steady trajectory predictions.

As mentioned above, this moving-body process is controlled by a run script. At AEDC there is an ongoing effort under HPC CHSSI Project CFD-9[20] to build a generalized framework for moving-body problems. This framework gives the user the ability to "plug 'n play" different flow solvers, grid assemblers, and SIXDOF codes. It's also designed to make problem setup more user friendly and less error prone by allowing one-time parameter inputs and handling the passing of these parameters between the codes. The framework has already been demonstrated using different grid assemblers and flow solvers, including an unstructured overset grid assembler and flow solver.

5. Results

Figure 2 shows a JSF aircraft that was computed under the current effort. The computations were performed on the full aircraft for a configuration that consisted of six externally mounted missiles and a gun pod and is typical of the complexity of the computations performed. The grid system consisted of 18×10^6 points distributed over 124 grids, and it required approximately 1760 CPU hours to converge to a steady-state solution. Although the convergence time was dependent on flow conditions, the grid size and resource requirement were typical for all full aircraft computations. The full aircraft solutions were performed only for nonzero sideslip angles. For zero sideslip cases, flow symmetry was assumed to be at the aircraft center plane, and the grid size was basically halved. This particular case was run to determine carriage loads, which were supplied to the structural analysis group at Lockheed Martin (LM). Additional computations were performed to determine carriage load for other loadouts. These loadouts included both internally and externally mounted stores on the Air Force as well as the Navy variants. For internal weapons carriage loads the computational resources usually increase. This is because the bay flow is inherently unsteady, and additional iterations are required after the startup transients dissipate to determine time-averaged loads.

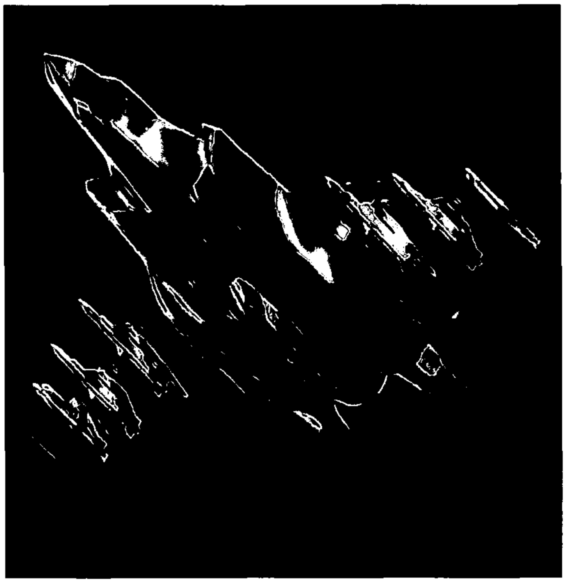

Figure 2. JSF Carriage Load Solutions

Computations were performed at flow conditions both within and beyond the wind tunnel operating envelope. Conditions beyond the operating envelope of the tunnel involved computations at high Mach number. These Mach numbers were at or above operational requirements for the aircraft. Likewise, computations were performed for high angles of attack to determine the loads at the boundaries of the operational requirements. After the test the computations performed for flow conditions within the operating envelope were compared to the wind tunnel data at matching conditions. Excellent agreement was achieved between computational carriage loads and wind tunnel data. This agreement provided confidence to use the computed results for conditions outside the operating envelope of the tunnel. This strategy provided carriage loads for conditions that would have required additional wind tunnel tests, thus saving the program millions of dollars.

Additional computations were performed for the JSF weapon separation. Fully time-accurate moving-body computations were performed to simulate the release trajectory of several weapons, including both internally and externally mounted stores. The complexity of these computations was increased by releasing the store from a maneuvering aircraft as discussed above. Another set of computations involved the release trajectory simulation of fuel tanks with different fuel states. The constrained motion off the pivot mechanisms was modeled, thus requiring some additional modifications to the SIXDOF program. Comparisons with CTS wind tunnel data showed excellent agreement even for constrained release of stores for maneuvering aircraft. For tank releases from the pivot mechanism, the tank will tumble. Because of the constraints of the CTS support system in the wind tunnel, the trajectory is normally terminated before it is evident that the tank will clear the aircraft. The time-accurate CFD simulations allow the continuation of the trajectory until a clearance judgement can be made. This is an added risk reduction achieved by using this type of analysis. Additional savings to the JSF wind tunnel test program were realized from the ability of the time-accurate CFD computations to evaluate candidate configurations prior to the test, thus enabling narrowing of the list of tested configurations to those showing the best probability of success.

The J-UCAS effort primarily involved the computations of fixed-position stores and flow fields for use with engineering methods in performing trajectory analyses of various weapons from the weapons bays. However, a number of other computations were performed during the planning phase of the wind tunnel test. The existing store separation test article was large, and this led to concerns about potential compromising effects of wind tunnel wall interference on the store separation data. The customer was faced with additional cost of testing in a larger facility or possibly fabricating another, smaller test article to make sure wall interference would not be an issue. Utilizing CFD and a model of the porous walls in the wind tunnel[21], computations were performed in the tunnel and in free air. A store separation analysis was performed using the aforementioned engineering methods, and it showed that the separation data would not be compromised by the scale of the test article. With the computational processes, confidence was achieved that high-quality results could be obtained with the existing test article model while saving the program several hundred thousand dollars in additional test costs that otherwise might have been spent in a larger facility or on fabricating another test article. Pretest computations were also performed to assess weapons bay unsteadiness and to analyze several spoiler candidates and their effects on weapon bay dynamics. However, the primary purpose of the pretest computations was to determine the separation of the several stores using quasi-steady, semiempirical engineering methods where CFD supplied all information necessary to perform the analysis except the freestream database. This involved computing the flow field below the aircraft without the active store in the flow field and computing the active store with the aircraft at two positions near carriage. As mentioned in previous sections, the latter computations are used to determine the decay in the mutual interaction between the parent and the store. Comparison of the wind tunnel and pretest trajectories is in progress.

The typical grid system for the J-UCAS with store consisted of approximately 6 million points distributed over 30 meshes. Because of the unsteady nature of the bay flow, the computations required enough iterations to determine time-averaged loads for the followup analysis with the engineering methods. Typical runs required about 2400 CPU hours.

Additional computations and separation analysis were performed on a previous version of the aircraft tested in the Propulsion Wind Tunnel facility's Aerodynamic Wind Tunnel 4T, at AEDC. The previous version is now a flight demonstrator and recently underwent flight separation testing. Since AEDC had previously performed computations on this demonstrator and had available computational resources, AEDC was able to quickly perform these computations and analysis. These computations were actually performed prior to, and in support of, this flight test and provided confidence that the store would separate cleanly. The computations and analysis were a major contribution to the flight test program success.

6. Conclusions

Computational support using HPC assets has provided vital results to two programs: the JSF and the J-UCAS. When possible, computations were validated against wind tunnel data, and the validations were so successful that entire tests, scheduled in other facilities to obtain data outside the operating envelope of Tunnel 4T, were not necessary. Elimination of these tests has saved the two programs millions of dollars. The subject computations also have: 1) provided information for pretest planning; 2) corrected errors in wind tunnel data reduction and results; and 3) obtained results for configurations that could not be obtained in the wind tunnel because of physical limitations of the support systems as well as limitations of the tunnel operating envelope. This project has truly shown the strength of the Integrated Test and Evaluation process in reducing cost and risk.

Acknowledgments

The research reported herein was performed by the Arnold Engineering Development Center (AEDC), Air Force Materiel Command. Work and analysis for this research were performed by personnel of Aerospace Testing Alliance, the operations, maintenance, information management, and support contractor for AEDC. Further reproduction is authorized to satisfy needs of the US Government.

References

1. Tramel, R.W. and R.H. Nichols, "A Highly Efficient Numerical Method for Overset-Mesh Moving-Body Problems." *AIAA Paper No. 97-2040*, AIAA 13th Computational Fluid Dynamics Conference, Snowmass Village, CO, June 29–July 2, 1997.

2. Sickles, W.L., C.H. Morgret, A.G. Denny, and R.H. Nichols, "Comparison of Time-Accurate CFD and Engineering Methods for the JDAM Separation from an F-18C Aircraft." 2001 Aircraft-Store Compatibility Symposium sponsored by International Test and Evaluation Association, Joint Ordinance Commanders Group, Aircraft Store Compatibility Subgroup, Destin, FL, March 5–8, 2001.

3. Sickles, W.L., A.G. Denny, and R.H. Nichols, "Time-Accurate CFD Store Separation Predictions with Comparison to Flight Data." *AIAA Paper No. 2000-0796*, 37th Aerospace Sciences Meeting and Exhibit, Reno, NV, January 10–13, 2000.

4. Carman, Jr., J.B., D.W. Hill, Jr., and J.P. Christopher, "Store Separation Testing Techniques at the Arnold Engineering Development Center, Volume II: Description of Captive Trajectory Store Separation Testing in the Aerodynamic Wind Tunnel (4T)." *AEDC-TR-79-1*, Vol. II (AD-A087561), June 1980.

5. Keen, K.S., "New Approaches to Computational Aircraft/Store Weapons Integration." *AIAA Paper No. 90-0274*, AIAA 28th Aerospace Sciences Meeting, Reno, NV, January 8–11, 1990.

6. Fox, J.H., T.L. Donegan, J.L. Jacocks, and R.H. Nichols, "Computed Euler Flowfield for a Transonic Aircraft with Stores." *Journal of Aircraft*, Vol. 28, No. 6, June 1991, pp. 389–396.

7. Donegan, T.L. and J.H. Fox, "Analysis of Store Trajectories from Tactical Fighter Aircraft." *AIAA Paper No. 91-0183*, AIAA 29th Aerospace Sciences Meeting, Reno, NV, January 7–10, 1991.

8. Sickles, W.L., M.J. Rist, C.H. Morgret, S.L. Keeling, and K.N. Parthasarathy, "Separation of the Pegasus XL from an L-1011 Aircraft." *AIAA Paper No. 94-3454*, AIAA Atmospheric Flight Mechanics Conference, August 1–3, 1994.

9. Morgret, C.H., R.E. Dix, and L.E. Lijewski, "Development of Analytical and Experimental Techniques for Determining Store Airload Distribution." *Journal of Spacecraft and Rockets*, Vol. 19, No. 6, November–December 1982, pp. 489–495.

10. Benek, J.A., J.L. Steger, F.C. Dougherty, and P.G. Buning, "Chimera: A Grid-Embedding Technique." *AEDC-TR-85-64* (AD-A167466), April 1989.

11. Jordan, J.K., N.E. Suhs, R.D. Thoms, R.W. Tramel, J.H. Fox, and J.C. Erickson, Jr., "Computational Time-Accurate Body Movement Methodology, Validation, and Application." *AEDC-TR-94-15*, October 1995.

12. Einfeldt, B., "On Gudunov-Type Methods for Gas Dynamics." *SIAM Journal of Numerical Analysis*, Vol. 25, No. 2, April 1988, pp. 294–318.

13. Einfeldt, B., C.D. Munz, P.L. Roe, and B. Sjogreen, "On Gudunov-Type Methods Near Low Densities." *Journal of Computational Physics*, Vol. 92, No. 2, 1991, pp. 273–295.

14. van Leer, B., "Towards the Ultimate Conservative Difference Scheme V. A Second Order Sequel to Gudunov's Method." *Journal of Computational Physics*, Vol. 32, 1979, pp. 101–136.

15. Nichols, R.H., "Development and Validation of a Two-Equation Turbulence Model with Wall Functions for Compressible Flow." *AIAA Paper No. 96-2385*, AIAA 14th Applied Aerodynamics Conference, New Orleans, LA, June 17–20, 1996.

16. Spalart, P.R. and S.R. Allmaras, "A One Equation Turbulence Model for Aerodynamic Flows." *AIAA Paper No. 92-0439*, AIAA 30th Aerospace Sciences Meeting and Exhibit, Reno, NV, January 6–9, 1992.

17. Menter, F.L. and C.L. Rumsey, "Assessment of Two-Equation Turbulence Models for Transonic Flows." *AIAA Paper No. 94-2343*, June 1994.

18. Suhs, N.E., and R.W. Tramel, "PEGSUS 4.0 Users Manual." *AEDC-TR-91-8* (AD-A280608), November 1991.

19. Chan, William M. and Pieter G. Buning, "User's Guide for FOMOCO Utilities — Force and Moment Computation Tools for Overset Grids." *NASA TM 110408*, July 1996.

20. Power, G.D., "An Update on the CHSSI CFD-9 Project: Unstructured Overset Grid Framework for Moving Body Applications." Paper presented at the HPCMP Users Group Conference, Williamsburg, VA, June 8–10, 2004.

21. Martin, Jr., F.W., W.L. Sickles, and S.A. Stanley, "Transonic Wind Tunnel Wall Interference Analysis for the Space Shuttle Launch Vehicle." *AIAA Paper No. 93-0420*, AIAA 31st Aerospace Sciences Meeting, Reno, NV, January 11–14, 1993.